

A matheuristic algorithm for the three-dimensional loading capacitated vehicle routing problem (3L-CVRP)



Un algoritmo híbrido para el problema de ruteo de vehículos con restricciones de carga de tridimensional

Luis Miguel Escobar-Falcón¹, David Álvarez-Martínez², Mauricio Granada-Echeverri¹, John Willmer-Escobar³, Rubén Augusto Romero-Lázaro²

¹Facultad de Ingenierías, Universidad Tecnológica de Pereira. Carrera 27 #10-02 Barrio Álamos. A. A. 097. Pereira, Risaralda, Colombia.

²Departamento de Ingeniería Eléctrica, Universidad del Estado de São Paulo. Campus de Ilha Solteira, Av. Professor José Carlos Rossi, 1370-Campus III. CEP: 15385-000. Ilha Solteira, Brasil.

³Departamento de Contabilidad y Finanzas, Facultad de Ciencias de la Administración, Universidad del Valle. Calle 4B # 36-00, Sede San Fernando, Edificio 124. A. A. 25360. Cali, Colombia.

ARTICLE INFO

Received December 2, 2014

Accepted January 29, 2016

KEYWORDS

Vehicle Routing Problem, Matheuristics, Branch-and-Cut, packing

Problema de enrutamiento de vehículos, Matheurísticas, Branch-and-Cut, empaquetamiento

ABSTRACT: This paper presents a hybrid algorithm for solving the Capacitated Vehicle Routing Problem with practical three-dimensional loading constraint. This problem is known as 3L-CVRP (Three-dimensional Loading Capacitated Vehicle Routing Problem). The proposed methodology consists of two phases. The first phase uses an optimization procedure based on cuts to obtain solutions for the well-known Capacitated Vehicle Routing Problem (CVRP). The second phase validates the results of the first phase of a GRASP algorithm (Greedy Randomized Adaptive Search Procedure). In particular, the GRASP approach evaluates the packing constraints for each performed route of the CVRP. The proposed hybrid algorithm uses a relaxation of the classical model of two sub-indices for the vehicle routing problem. Specifically different types of cuts are added: subtour elimination, capacity-cut constraints, and packing-cut constraints. The proposed algorithm is compared with the most efficient approaches for the 3L-CVRP on the set of benchmark instances considered in the literature. The computational results indicate that the proposed approach is able to obtain good solutions, improving some of the best-known solutions from the literature.

RESUMEN: En este artículo se presenta un algoritmo híbrido para resolver el problema de ruteo de vehículos con restricciones de capacidad y restricciones prácticas de empaquetamiento tridimensional, este problema en la literatura es conocido como 3L-CVRP (Capacitated Vehicle Routing Problem and Container Loading Problem). La metodología de solución propuesta en este trabajo consiste de dos fases. La primera utiliza un procedimiento de optimización basado en cortes para el Problema de Rutas de Vehículos Capacitados (CVRP). La segunda valida las soluciones de la fase anterior a través de un algoritmo GRASP (Greedy Randomized Adaptive Search Procedure), el cual evalúa las restricciones de empaquetamiento de cada una de las rutas. Para el algoritmo híbrido se utiliza la relajación del modelo clásico de dos subíndices para el problema de ruteo de vehículos. En particular diferentes tipos de cortes son adicionados: eliminación de subtours, cortes debido a las restricciones de capacidad y cortes para restricciones de empaquetamiento. El algoritmo propuesto ha sido comparado con los algoritmos más eficaces para el 3L-CVRP en el conjunto clásico de instancias presentadas en la literatura. Los resultados computacionales muestran que el método propuesto es capaz de obtener buenos resultados perfeccionando algunas de las mejores soluciones conocidas propuestas en la literatura.

* Corresponding author: Luis Miguel Escobar Falcón

e-mail: luismescobarf@gmail.com

ISSN 0120-6230

e-ISSN 2422-2844



1. Introduction

Many actions in the transit of products involve two problems, which have been studied deeply in the last decades. The first problem is referred to the design of the routes to fulfill the demand of the customers by considering the minimum travelling cost (Capacitated Vehicle Routing Problem-CVRP). The second problem considers the best way to load the products in the used vehicles for the performed routes (Three-dimensional Container Loading Problem - 3D-SLOPP). These problems belong to the well-known NP-hard problems for which the solution is really challenging.

The Vehicle Routing Problem (VRP) arises in the distribution of a set of products to a number of customers who are geographically dispersed, by regarding the minimization of the distribution costs or the maximization of the net income associated with the transportation. The Three-Dimensional Container Loading Problem or Three-Dimensional Knapsack problem seeks to accommodate a number of elements within a rectangular box (container) by considering several objective functions and by satisfying determined packing constraints. The combination of both problems has several realistic applications in many industrial contexts such as the transportation of chipboard for furniture, the delivery of courier companies, the conveyance of vehicles, and the transit of products on pallets, among others.

The combined problem of routing and packing considered is a variant of the well-known routing problem called Capacitated Vehicle Routing Problem (CVRP) and the variant of the packing problem called Container Loading Problem (3D-SLOPP). The CVRP seeks to perform a set of routes starting and ending at a central depot. The CVRP could be defined as a set of K homogeneous vehicles (each one having a capacity Q), which must satisfy the demand of a set of N customers. Each vehicle is assigned to at least one route. A single vehicle must visit each customer one time. The sum of the demands of the customer visited on a single route must not exceed the vehicle capacity Q . The objective is to minimize the sum of the traveling cost for the performed routes.

The 3D-SLOPP must be solved for each performed route. The 3D-SLOPP consists in loading a set of small boxes B inside a container. The set B has different sizes and limited amounts. The objective is the maximization of the available occupied space. This problem is also well known as the three-dimensional knapsack problem (3D-SKP) or Three-Dimensional Single Large Object Packing Problem (3D-SLOPP). For the 3D-SLOPP, the orientation and the fragility of the boxes, the load stability, and the sequence of the load must be considered. Several variants of the 3D-SLOPP are obtained by considering different type of packing constraints. In this work, we also study the variant of the 3D-SLOPP problems by taking into account that the vehicle capacity Q is equal to its volume, which is attractive for many real applications of the industry. The integration of both problems arises to the well-known problem 3L-CVRP, i.e. Three-dimensional Loading Capacitated Vehicle Routing Problem.

In this work, we have proposed a matheuristic algorithm, which is computationally compared with the most effective heuristics for the 3L-CVRP. The new proposed approach obtains competitive results on the classical set of benchmarking instances for the 3L-CVRP. The main contribution of this paper is to propose a new successful matheuristic approach for the 3L-CVRP by considering a combination of exact techniques with a GRASP approach which guarantees the loading constraints. The proposed algorithm is a novel matheuristic approach which combines a GRASP approach with exact algorithm for getting good results. While a combination between exact techniques and Tabu Search (TS) has been proposed in the literature for the CVRP (see e.g. [1]) no attempt has been proposed for combining exact techniques with a GRASP scheme for the 3L-CVRP. The former algorithm is able to improve the best-known solutions found by the most effective published algorithms on a set of instances taken from the literature.

The paper is organized as follows. The literature associated to the Capacitated Vehicle Routing Problem and the Packing Problem is described in Section II. Section III gives a formal definition of both problems into the 3L-CVRP and the literature proposed to solve it. Section IV presents a detailed description of the framework used by the proposed algorithm. A computational comparative study on the classical set of benchmark instances from the literature is provided in Section V. Finally, Section VI contains concluding remarks and future research.

2. Literature review

The vehicle routing problem considering load constraints is a relative new interesting research subject. The interest of researchers and practitioners is motivated by the intrinsic difficulty in this area, which combines two NP-hard problems: the Capacitated Vehicle Routing Problem and the Container Loading Problem.

2.1. Capacitated Vehicle Routing Problem (CVRP)

The Capacitated Vehicle Routing Problem (CVRP) seeks to find a specified number of cycles (routes) to fulfill the demand of a set of vertex (customers) by starting and finishing at a central depot located in the vertex 0. A complete formulation of the CVRP proposed in [2], which is well known as formulation of two indices is shown as follows:

The CVRP could be described as graph theory problem. Let $G = (V, A)$ be a complete graph, where $V = \{0, 1, 2, \dots, n\}$ is the vertex set and A is the arc set. Vertices $V = \{0, 1, 2, \dots, n\}$ correspond to the customers, whereas vertex 0 corresponds to the depot. A non-negative traveling cost c_{ij} , is associated with each arc $(i, j) \in A$. The traveling cost between $[i, i]$ is not allowed. Therefore, the cost $c_{ii} = +\infty$ for all $i \in V$. In particular, this paper considers the symmetric version of the CVRP (SCVRP). Therefore, $c_{ij} = c_{ji}$ for all $(i, j) \in A$, and the arc set could be replaced by a complete set of undirected

edges, E . Each vertex $i \in V$ is associated with a known nonnegative demand, d_i , to be delivered. Note that the depot has a fictitious demand $d_0 = 0$. Given an edge $e \in E$, let $\alpha(e)$ and $\beta(e)$ denote its endpoint vertices. Given a vertex set $S \subseteq V$, let $\delta(S)$ and $E(S)$ denote the set of edges $e \in E$ that have only one or both endpoints in S , respectively. In addition, let $d(S) = \sum_{i \in S} d_i$ be the total demand of the set S .

A set of K identical vehicles, each with capacity Q , is available at the depot. To ensure feasibility we assume that $d_i \leq Q$ for each $i = 1, \dots, n$. Each vehicle performs only one route. For a set $S \subseteq V \setminus \{0\}$, we denote by $r(S)$ the minimum number of vehicles needed to serve all customers in S . Often, $r(S)$ is replaced by the trivial Bin Packing Problem lower bound $\lceil \frac{d(S)}{Q} \rceil$. The BPP allows determining the minimum number of bins (vehicles), each one with capacity Q , required to load all the n items, each with nonnegative weight $d_i, i = 1, \dots, n$, being NP-hard in the strong sense.

The CVRP consists of finding a set of K performed routes (each one corresponding to one vehicle) with minimum cost, defined as the sum of the costs of the arcs belonging to the performed routes. The CVRP is subject to the following constraints

- i. Each route ends and begins in the depot vertex
- ii. Each customer vertex is visited by exactly once; and
- iii. The sum of the demands of the vertices visited by a route must not exceed the vehicle capacity, Q .

The model employed in this paper, is a two-index vehicle flow formulation that uses binary variables x to indicate if a vehicle travels an arc (i, j) in the optimal solution (7). In other words, variable x_{ij} takes value 1 if arc $(i, j) \in E$ belongs to the optimal solution and takes value 0 otherwise. The objective function is to minimize the cost of the traveled arcs (1). The Eqs. (2-6) control the visits to the clients and the subtour elimination.

$$\text{minimize } \sum_{i \in V} \sum_{j \in V} c_{ij} x_{ij}; \tag{1}$$

subject to

$$\sum_{i \in V} x_{ij} = 1 \quad \forall j \in V \setminus \{0\} \tag{2}$$

$$\sum_{j \in V} x_{ij} = 1 \quad \forall i \in V \setminus \{0\} \tag{3}$$

$$\sum_{j \in V} x_{0j} = K \tag{4}$$

$$\sum_{i \in V} x_{i0} = K \tag{5}$$

$$\sum_{i \notin S} \sum_{j \in S} x_{ij} \geq r(S) \quad \forall S \subseteq V \setminus \{0\}, S \neq \emptyset \tag{6}$$

$$x_{ij} \in \{0,1\} \quad \forall i, j \in V \tag{7}$$

Constraints (2) and (3) correspond to the flow constraints of the set V (customers) i.e., these constraints guarantee that each customer must be visited by one single vehicle only once. Constraints (4) and (5) ensure that the same number of routes arrive and leave the depot 0. Eq. (6) consider the subtour elimination constraints. According to [2], this set of constraints could be interpreted as follows in the Eq. (8):

$$\sum_{i \in S} \sum_{j \in S} x_{ij} \leq |S| - r(S) \tag{8}$$

However, the set of subtour elimination constraints require special considerations due to its combinatorial complexity when the number of customers is increased. Therefore, we have considered the set of constraints (8) for the proposed algorithm. In particular, we added the required cuts to eliminate subtour until a feasible solution is found during the branch and bound procedure.

Note that the considered model is able to represent only the CVRP problem. For getting a global representation of the 3L-CVRP problem, we have considered the set of constraints (9), which are added iteratively together with the subtour elimination constraints. This set of constraints guarantee the feasibility of the packing requirements (such as multi-drop constraints, among others). Let $U \subseteq V \setminus \{0\}$ be a subset of customers with a cumulative demand, which cannot be packed in the vehicles by considering the sequential packing or multi-drop constraints. All the possible subsets of customers who can not be packed, are controlled by constraints (9):

$$\sum_{i \in U} \sum_{j \in U} x_{ij} \leq |U| - 2 \tag{9}$$

By adding this set of constraints, the mathematical model could represent properly the 3L-CVRP problem. The level of difficulty of the solution of the model (1) - (9) increases when the number of nodes increases, due to the combinatorial explosion of possible subsets of customers controlled by Eq. (8). The high complexity of the CVRP has led the development of the various algorithms on exact and approximate methods. The CVRP has been investigated since the decade of the 50's. Reviews of the CVRP are presented in [2-4]. In [5], a vehicle routing problem arising in supply chain management is proposed (including the 3L-CVRP).

2.2. Loading and Backing Problem

In realistic loading and packing problems, the demand of the customers is not simply characterized by a quantity (as in the case of CVRP), but it also is determined by its shape and location in the space. In this case, it is necessary to ensure that an item to be carried on must be placed into the space used by a vehicle. These constraints are concerned with the multidimensional rectangular packing problems, which originate as an extension of the one-dimensional Bin Packing Problem (BPP). The BPP can be described as the problem of placement of a set of segments without overlapping. A general introduction to the rectangular packing research area is given by [6-9].

3. Problem definition

The 3L-CVRP considers that three-dimensional items generate the full weight of demand of customer i ($i = 1, 2, \dots, n$). Each item I_{il} ($l = 1, 2, \dots, m$), has a width w_{il} , a high h_{il} and a length l_{il} . The loading surface of each vehicle has a width W , a high H and a length L . Let $S(k) \subseteq \{1, 2, \dots, n\}$, be the set of customers visited by the vehicle K . The 3L-CVRP imposes a packing constraint without any variety of three-dimensional overlap of all items ordered by each customer $S(K)$ within the cargo space of dimensions $W \times H \times L$. The packing constraints for the 3L-CVRP are characterized by the following aspects:

- Orientation: Items have a fixed orientation or can be rotated 90° in the horizontal plane by keeping off the rotation of the vertical orientation.
- Fragility: An item I_{il} could have a fragility f_{il} . If f_{il} is equal to 1, I_{il} is fragile, and 0 otherwise. In this case, non-fragile items cannot be placed over fragile items.
- Area support: Each item I_{il} is packaged over other items. Let \bar{A} be the area of the bottom of product I_{il} . The packaging is feasible only if $\bar{A} \geq \alpha w_{il} l_{il}$, where α is a given threshold ($0 \leq \alpha \leq 1$) and represents the minimum portion of the area of the box, which is contacted with box of item I_{il} (item for which the current box is supported).
- Sequential load: When an item is unloaded, there must be a chronological succession of straight movements in the direction of the rear of the vehicle, allowing the process of unloading without moving any other item. In other words, any item requested after customer i may be placed on I_{il} or between I_{il} and the backside of the vehicle.

This work considers all the packing constraints by characterizing them as previously expressed. It is worth to note that this characterization considers many assumptions limiting the functionality and applicability on a real context. For example, the constraint of fragility could be formulated as a binary expression depending of the load-bearing strength and the orientation of the boxes [10].

The previous published works for the 3L-CVRP have tried to eliminate some of the packing constraints in order to distinguish the most critical. In [11], the authors indicate that the loading sequence is usually the dominant constraint. In this work, other features of the problem are studied; particularly, the fact of the vehicle capacity Q is usually specified as a parameter without any relation to the type of load. Therefore, this aspect implies that the value of $d(i)$ (demand for each customer) assumes that all the boxes for the set of customers have the same density of material with different demand; i.e. each customer has different types of boxes with different density but all the boxes must have the same density.

There are two type of assumptions respect to the vehicle capacity constraints:

- Transportation of all the type of material density of boxes for each customer
- Transportation of one type of material density of the boxes (which is assumed as 1, being the weight of each box equal to its volume).

The previous published works only consider the first assumption. We have considered both assumptions (see results of Matheuristic column and Matheuristic (3L-VRP) column in Table 1). Indeed, one contribution of the proposed work is to examine the effect of the density equal to 1. Consequently, for the second assumption, the new value of $d(s)$ will be $d(s)$ will be $d(s) = \sum_{i \in s} w_{il} l_{il} h_{il}$ and the new vehicle capacity $Q = W \times L \times H$.

All the set of packing constraints are modeled and linked with the transportation model in [12], but due to the high complexity of the model, the proposed methodology is inadequate for solving medium and large size problems. In this work, a GRASP algorithm within the mathematical model guarantees these constraints, allowing the solution of real problems of the companies. Several approximate algorithms have been proposed to solve the 3L-CVRP.

The problem of 3L-CVRP by considering time window constraints is proposed in [13]. In this work, several constructive heuristics are addressed. In [14], an extension of the previous described work [13] is presented. In particular, a multi objective scheme is proposed by considering the following objective functions: minimizing the number of vehicles, minimizing the total travel distance, and maximizing the used volume.

In [15], a Tabu Search algorithm is proposed for solving the 3L-CVRP. In this work, for each neighborhood solution of the vehicle routing problem, the payload is determined by another Tabu Search scheme for the Three-dimensional Strip Packing Problem (3SPP). If the resulting load exceeds the capacity of the vehicle, the solution is accepted by considering a penalization scheme. Other Tabu Search algorithm has been proposed by [16]. The loading problem is solved by a heuristic for the minimal waste of space. Finally, a tabu search algorithm solves the corresponding routing problem. [17] propose a generalization of the three-

dimensional for the bi-dimensional case. In [18], an Ant Colony Optimization (ACO) is presented for the 3L-CVRP. Heuristic approaches for which the vehicle routing problem is solved by metaheuristic algorithms based on population are given in [19, 20]. In [19], a bee metaheuristic algorithm with a Tabu Search is proposed for the 3L-CVRP. A heuristic algorithm for the loading problem with a genetic algorithm for the 3L-CVRP has been proposed in [20]. In [21], two heuristics for the packing problem are improved and introduced within a Tabu search scheme for solving the considered problem. Experiments computational show the efficiency of the proposed approach.

An uncapacitated 3L-CVRP is introduced in [22]. Two heuristic approaches to solve this variant of the original problem are proposed and compared. A two-stage heuristic for solving the problem considered in [13] is presented in [23]. The first stage optimizes the packing problem, while the second deals with the aspect of the corresponding routing problem. Computational experiments show the high efficiency of the method.

In [24], the author introduces an efficient hybrid approach based on a Tabu search algorithm for the vehicle routing subproblem. In the proposed approach, the generated routes are ordered in a list, which is sorted increasingly according to the travelling cost. For each solution in the resulting list, a tree search algorithm for solving the loading subproblem is performed. Computational experiments show the effectiveness of the proposed methodology.

Note that all the proposed approaches proposed for the 3L-CVRP are based on heuristic schemes except for [24]. In this paper, we propose a matheuristic algorithm, which differs from [24] because the routing problem is solved by an exact method and the packing problem by an approximate algorithm. The proposed algorithm is explained in the following sections.

4. Matheuristic approach

The general solution strategy proposed addresses both problems separately: the Capacitated Vehicle Routing Problem (CVRP) and the Three Dimensional Container Loading Problem (3D-SLOPP). In particular, for each solution of a CVRP, a validation of the packing constraints of the cargo of the containers for each route is performed. The main strength of the proposed approach is that the computational effort is mainly focused on the exact solution for the CVRP, while the loading problem is solved by a GRASP approach of high performance. The GRASP approach is calibrated according to the characteristics of the items to be delivered, i.e., the cumulative demand of the customers covered by each route.

The mathematical model of two-index (1-9) is relaxed by eliminating the capability and the subtour elimination constraints. The proposed approach gradually inserts these constraints during the branch and cut scheme in order to obtain feasible solutions. Indeed, the proposed

algorithm begins with an initial solution generated by the well-known Clark & Wright algorithm and validated by the GRASP algorithm. The objective function value of the initial solution is used as upper bound of the proposed approach.

The algorithm allows infeasible solutions for the 3L-CVRP due to that the first feasible solution found during the search corresponds to the optimum of the problem. In particular, when a feasible solution for the routing problem is found, it provides a lower bound for the original problem. Therefore, it can be used as initialization for the 3L-CVRP during the search. The upper bound is given by the corresponding load demand for the K performed routes (the problem is studied as a bin-packing). However, the effort of the proposed algorithm is totally oriented to find feasible solutions for the vehicle routing problem guided by the minimization of total travelling cost. Although the two-index model (1-9) requires a notable computational effort, it allows articulating properly the constraints related with the management of the loading of the boxes in the container.

The 3L-CVRP has been addressed by applying an exact method (branch-and-cut) for the vehicle routing problem solved by ILOG Concert Technology and CPLEX. The packing problem is solved by a GRASP approach. Initially, the relaxed version of the model (1-9) is solved without the capacity and elimination of subtour constraints. Then, these constraints are added iteratively to the branch and cut process together with the packing constraints during the search procedure. The algorithm finishes when a solution is found for the model involving all the constraints. The matheuristic algorithm is described as follows:

4.1. Exact method for the Vehicle Routing Problem

In the proposed algorithm, we have considered the two-index model (1-9) to solve the subproblem of CVRP. This formulation is based on subsets of S customers to control the appearance of subtours and the capacity constraints of the vehicles. Constraints (6) or (8) are eliminated obtaining a relaxed model (M_{CVRP}).

In particular, the proposed approach starts by solving the M_{CVRP} and by keeping its optimal solution in S . Then, the current optimal solution for the M_{CVRP} is checked to find violations of capacity constraints (L_u) or the presence of subtours (L_s). If this solution is infeasible, the corresponding cuts are added to avoid infeasibilities by the function *AddCutsToTheModel* (M_{CVRP}, L_s, L_e, R_i). If the solution is feasible, the procedure must check the packing of demand for each route by a GRASP approach, which determines whether it is possible to pack the boxes into the vehicles with a pattern of loading without rearrangement. In addition, if the inverse route (inverse position of the customers) is impossible to pack, the considered routes must be prohibited (L_u). The pseudocode of the proposed algorithm is detailed in Figure 1.

Data: CVRP Mathematical Model M_{CVRP} , Initial solution with a modified Clarke&Wright Savings (packing feasible) R_0 , 3L-CVRP Solution R_s , Subtours List L_s , List of routes with unfeasible packing L_u , Time of execution for the incremental model in seconds T_e

Result: Optimal and feasible solution R_s for the 3L-CVRP

$R_0 \leftarrow$ ModifiedSavings;
 Initialize mathematical model M_{CVRP} without the constraints with combinatorial explosion;
 Start timer T_e ;
while ($L_s \neq \emptyset$ or $L_u \neq \emptyset$) and $T_e < 3600$ **do**
 $L_s \leftarrow \emptyset, L_u \leftarrow \emptyset$;
 $R_s \leftarrow$ Solve(M_{CVRP} , Starting point R_0);
 $L_s \leftarrow$ GetSubtours(R_s);
 if $L_s = \emptyset$ **then**
 for Each Route r of R_s **do**
 $L_u \leftarrow$ GRASP(r);
 end
 end
 AddCutsToTheModel(M_{CVRP}, L_s, L_u);
end
return R_s ;

Figure 1 Matheuristic Branch & Cut - GRASP Algorithm

In order to avoid the appearance of routes violating the packing constraints, the cuts $\sum_{i \in U} \sum_{j \in U} x_{ij} \leq |U| - 2$ (9) are added to the model M_{CVRP} . The edges of the routes with customers belonging to U (subset of customers of the infeasible routes by packing constraint) on a determined sequence will be restricted. However, several sequences of the customers are evaluated for a given route seeking to find feasible solutions by considering packing constraints. Figure 2 shows an example of the performed packing cuts:

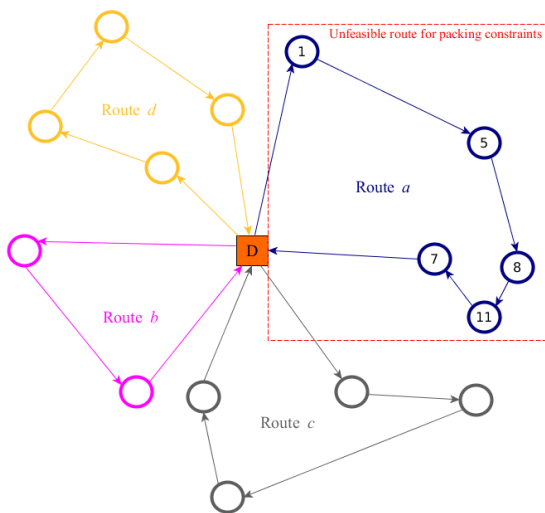


Figure 2 Solution obtained with the relaxed model

A solution for the relaxed M_{CVRP} could consider subtours and also routes that are unfeasible respect to the packing constraints (see Route a in Figure 2). The customers

belonging to Route a cannot be packed by the violation the sequential loading constraints, i.e., the route 0 - 1 - 5 - 8 - 11 - 7 - 0 must be eliminated. However, several permutations of Route a (example 0 - 7 - 11 - 8 - 5 - 1 - 0) are examined in order to check feasible solutions for the packing constraints. If any permutation of the route is possible to pack, its edges are considered for the next iterations of the Matheuristic. Therefore, the customers belonging to the unfeasible Route a make up the set $U = \{1, 5, 8, 11, 7\}$, $U \subseteq V \setminus \{0\}$ and $|U| = 5$. Then, according to the Eq. (9), the following cut (10) is applied:

$$x_{15} + x_{58} + x_{811} + x_{117} \leq 5 - 2$$

$$x_{15} + x_{58} + x_{811} + x_{117} \leq 3 \tag{10}$$

Consequently, any subsets of those edges are allowed, but the complete route sequence in the initial order is restricted on the following iterations once the model M_{CVRP} applies the cut (10). This process iterates on feasible solutions for the vehicle routing problem, but infeasible for the three-dimensional container loading problem. The optimal solution is found when the entire load of the routes of the optimal solution for the CVRP is also feasible for the 3D-SLOPP.

4.2. GRASP approach

In this work, we have considered an adapted version of the GRASP (Greedy Randomized Adaptive Search Procedure) algorithm presented in [11]. The proposed approach is based on the representation of maximum spaces, which allows obtaining feasible solutions by the control of the generation and the upgrade of these spaces in the constructive phase. The GRASP satisfies the constraints of the orientation of the boxes, load-bearing strength, the limit of the weight, the stability of the load, and the multi-drop patterns (loading with multiple destinations).

The GRASP algorithm was developed by [25] to solve difficult combinatorial optimization problems. Different researches show its quality and its robustness [26]. GRASP is an iterative procedure that combines a constructive phase and an improvement phase. In the constructive phase, a solution is built step by step, by adding items. The improvement phase is iterative, greedy, random and adaptive. In the following subsections the constructive phase, the random strategy, the moves developed for the improvement phase, and the diversification process, are described. The proposed search strategy allows randomizing the choice of the type and number of boxes to be located at each maximal space. The improvement phase applies several moves (compression of loading boxes and the refilling process).

4.3. Constructive algorithm

The constructive algorithm is based on the idea proposed by [27] for the classic container-loading problem. The main difference respect to this algorithm is the consideration of vertical stability, which must be guaranteed by packing patterns with full support. In addition, a special treatment

to upgrade structures of the remaining available spaces is considered. This aspect is important due to the management of the empty spaces is not longer trivial by trying to remove some items.

The constructive algorithm is based on the utilization of the maximal spaces. In this case, each selected box is packed in a new space, creating three new maximal spaces (see Figure 3). The constructive algorithm uses an updated list (FS) of the maximal spaces and a list B that contains the boxes of the current customers, which are not ready to be packed. The steps of the constructive algorithm are defined below.

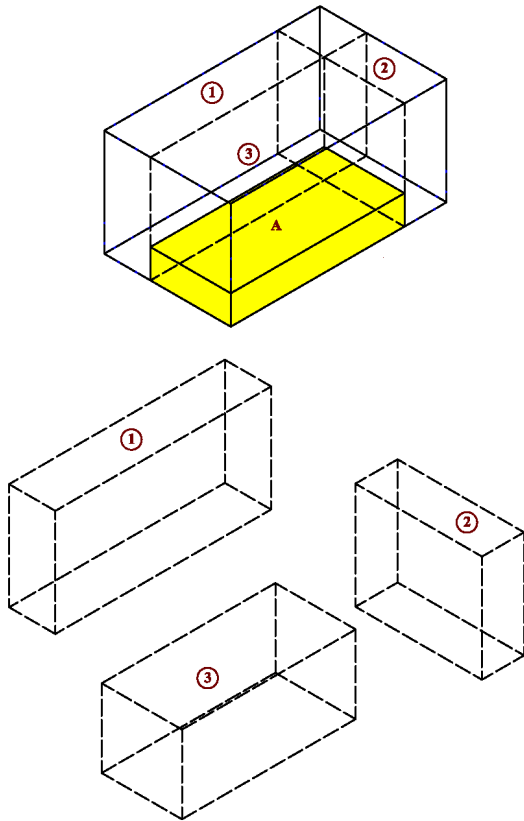


Figure 3 Box location into an empty space generating three maximal sub-spaces

Step 0: Initialization of FS . A list FS of empty spaces has been created for locating the selected boxes. Let $B = B_1, \dots, B_2, B_n$ be the list of the set of remaining boxes to be packed for a given customer.

Step 1: Choose a maximal space of FS . Since FS represents the maximum empty spaces with the largest available parallelepipeds to locate the boxes, it is necessary to determine a mechanism of selection of spaces based on some criteria of quality or packing strategies. In this paper, two criteria are proposed: choose the maximum space with the minimum distance to the backside of the container and choose the maximum space with the minimum distance to the roof of the container. In addition, the lower rear corner of the container is selected for the selected space as a reference to locate the boxes in the empty space.

Step 2: Select the boxes to be packed. Once the maximal space FS' has been selected, it is necessary to consider the sorted list B of the first box i that fits inside FS' . If there are multiple boxes type i , it is mandatory to generate each of the possible layers. Therefore, the boxes must be packed in arrays of columns or rows by combining the different axes. As in [27], two criteria are considered to select one of the configurations of boxes:

- Select the layer of boxes that produces the largest increase in the objective function (maximum volume). This is a greedy approach filling the space with the layer of a greater volume of the boxes.
- Select the layer of boxes that best fit in the maximum space. This is a criterion for which the distances between each side of the layer of boxes and each face of the maximum space are computed. The distances of each configuration are ordered in non-decreasing way for selecting the configuration with the lowest distance (which best fits to the space).

Step 3: Update the list FS . The packing process produces new maximal spaces FS' to be replaced in the list FS , except for the case either the box or the layer fits exactly in the space FS . Moreover, as the maximal spaces are not disjoint, the packed box (or the layer) can be intercepted with other maximal spaces, which could be reduced or eliminated.

The list FS is verified and updated once the new spaces have been added and some existing spaces have been modified. The list B is also updated and the maximal spaces that cannot locate of any of the boxes that still remain to pack must be removed from S . If $FS = \emptyset$ or $B = \emptyset$, this phase is finished. Otherwise, if there are still boxes to be packed of the current customer, the algorithm must return to step 1.

The list must be verified and the possible options be removed of the list, once the maximum spaces have been added and some existing spaces have been modified.

Step 4: Update the list S for a new customer. When the current customer has been packed, the maximal spaces must be updated depending of the criterion of multi-drop.

- Visible: The maximal spaces that are completely invisible from the door of the container must be removed from the list. In addition, the maximal spaces having a visible and an invisible part must be modified.
- Achievable: The unreachable spaces must be removed or updated.

4.4. Randomization

A layer is built according to the selected criterion (maximum volume or best fit) for each type of box and for each allowed orientation. Each layer is called as: configuration or candidate. When the full range of potential layers is constructed, a restricted list of candidates is

considered by selecting one of these layers randomly. We used a Restricted Candidate List (RCL) according to a determined value (this means that the candidates are sorted according to their quality value). If the value of the objective function of the candidate is greater than a threshold (δ), the candidate is located in the list RCL. In the process of building layers, we have considered the value C for each candidate, its lowest (C_{min}) value and its highest value (C_{max}). The candidate is accepted into the RCL only if it satisfies $C \geq C_{min} + \delta(C_{max} - C_{min})$. The parameter $\delta \in [0,1]$ controls the size of the list of candidates. If $\delta=0$, all the configurations are randomly selected and considered to the list. In contrast, $\delta=1$ indicates a completely greedy selection, because only the best candidate would be the only element in the list and it always be chosen. For values of $0 \leq \delta \leq 1$, the number of configurations in the list is not predefined, depending on the relative values of the candidates. In this case, δ takes randomly one value of 11 possibilities 0, 0.1, ..., 0.9 and 1. This value is selected depending on its performance in previous iterations. Indeed, if the value of δ has improved the current solution previously, the probability of its selection is increased and decreased otherwise.

4.5. Improvement

The improvement movement consists of eliminating the last $k\%$ boxes packed in the complete solution. We choose the value k at random from the interval [30, 90] as in [11]. The removed items plus the items that were left unpacked in the solution are then packed again using the deterministic constructive procedure guide by the objective function of Best-Volume. In this call of the deterministic algorithm, we can use the objective function Best-Volume. We consider that a solution has improved if the total volume of the packed boxes has increased.

The improvement phase is only called if the solution of the constructive phase is considered to be promising, that is, if it is considered a good starting point for improving on the best known solution. Therefore, we only consider those solutions that are above a certain threshold. At the beginning, the threshold takes the value of the first solution of the constructive algorithm. Then, if at an iteration the solution value is greater than the threshold, we update this threshold to this value and go to the improvement phase. If the solution value is lower than the threshold, the solution is not improved and the reject counter (niter) is increased. When the number of rejected solutions is greater than a value $maxFilter$, the threshold is decreased according to the expression:

$threshold = threshold - \lambda(1+threshold)$ where λ is set at 0.2 (as in [28]), and $maxFilter = 50\%$ total iterations.

Figure 4 shows the GRASP approach used to solve the packing subproblem of the proposed algorithm. The GRASP Algorithm begins by selecting one of the empty spaces (FS_i), and then the list of layers of boxes (C_s) that fit in the space FS_i is generated. Then, the list C is reduced to the Restricted Candidate List (RCL). One element is randomly selected of the list C_s . The layer C is located generating the pattern P

and forcing to update the lists of maximum spaces and the lists of the remaining boxes (FS and B , respectively). When all the boxes demanded by a customer ($B = \emptyset$) are assigned, the objective function is analyzed determining its quality. If it has good quality, the best solution found so far is updated. If there are still empty remaining spaces and there are customers for packing, then the maximum space list must be updated by changing customers (it should be necessary to eliminate the spaces that violate the multi-drop constraints). Finally, if there are no empty spaces, the process finishes.

Data: Lists of Spaces FS , List of Boxes B_c of each Customer c and List of Packing Patterns P

Result: Feasible or Unfeasible packing.

Initialize $FS = OriginalContainerSize$, $B = ListOfBoxesOfEachCustomer$, $P = \emptyset$

```

for  $l \leftarrow 1 \dots IterationsNumber$  do
  while  $S \neq \emptyset$  do
    while  $B_c \neq \emptyset$  do
      Space  $FS_i = SelectSpaceByCriteria(FS)$ ;
      List  $C_s = GenerateLayersList(FS_i, B)$ ;
      List  $RCL = BuildRCL(C_s, \delta)$ ;
      Layer  $C = SelectLayerRandomly(RCL)$ ;
      Packing Pattern  $P =$ 
      LocateLayer-Space( $C, FS_i, P$ );
      List  $FS =$ 
      UpdateListOfMaximalSpaces( $C, FS$ );
      List  $B =$ 
      UpdateListOfRemainingBoxes( $C, B$ );
    end
    List  $FS =$ 
    UpdateListOfSpacesByCustomerChanging( $FS, Multi-$ 
    dropType);
  end
  if  $f(P) > threshold$  then
    | Packing Pattern  $P = ImproveFinalPattern(P, k)$ ;
  else
    |  $threshold = threshold - \lambda(1 + threshold)$ 
  end
end
if  $B \neq \emptyset$  then
  | return Feasible;
end
return Unfeasible;

```

Figure 4 GRASP Algorithm

5. Computational results

We have considered the classical set of benchmark instances [27 instances] to validate the performance of the proposed methodology. The proposed matheuristic has been compared with [15, 17-19, 24, 29]. Several best known results have been improved. The computing time of the proposed methodology is quite high compared to the published approaches. The benchmark set for the 3L-CVRP has been taken from the library published in <http://or.dei.unibo.it/instances/three-dimensional-capacitated-vehicle-routing-problem-3l-cvrp>.

Table 1 shows the total cost of the performed routes to deliver all the boxes for the customers (objective function

of 3L-CVRP). The Figure 5 shows the routes obtained for the proposed algorithm. Figures 6-9 show the packing patterns corresponding to the routes of the solution for Instance 1. As is shown, the packing patterns satisfy the constraints of static stability, brittleness of the boxes and unloading without rearrangement (multi-drop or LIFO policy).

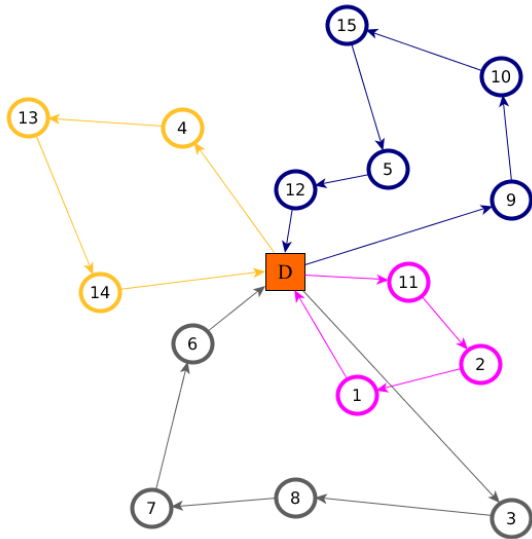


Figure 5 Routing for the first instance with feasible packing

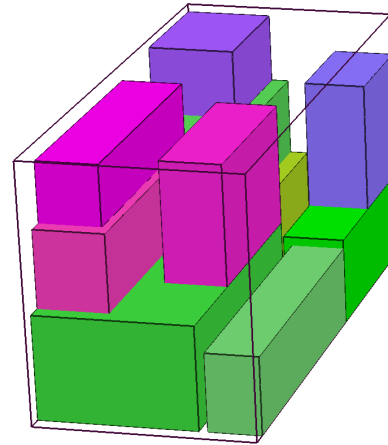


Figure 8 Route 9-10-15-5-12

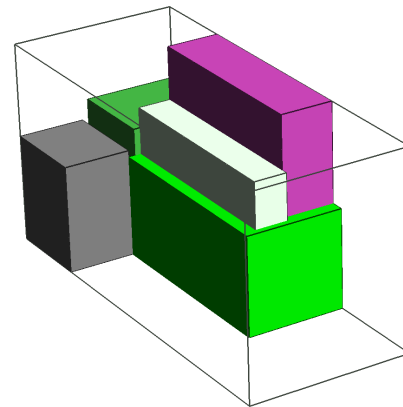


Figure 9 Route 11-2-1

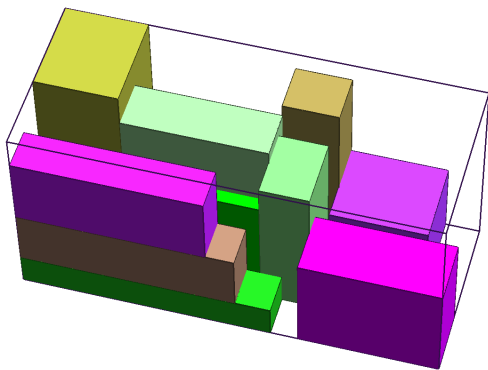


Figure 6 Route 3-8-7-6

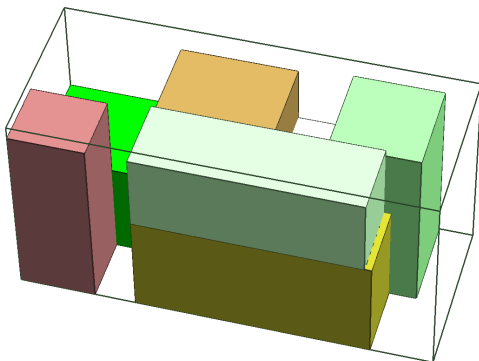


Figure 7 Route 4-13-14

The proposed methodology outperforms the quality of the solution found by the previous published algorithms presented in the literature. The computational times are high due to the use of exact method for the vehicle routing problem. The solutions obtained for the integrated 3L-CVRP problem by including packing constraints, make difficult the solution of the generation of routes to deliver products to the customers. The Figure 6 shows the performed routes for Instance 1. Note that some routes are not convex envelopes as in the traditional CVRP (see route located to the right in Figure 4). In addition, the performed routes clearly indicate a worse objective function for the CVRP. Indeed, the CVRP could be considered as a lower bound for the vehicle routing problem of the 3L-CVRP.

For the case of transportation of one type of structure of the boxes, the obtained results of the proposed approach respect to the packing constraints (see column Matheuristic (3L-VRP) in Table 1) are aligned with the conclusions of [10]. Indeed, the obtained solutions for the benchmarking set are at least of the same quality than the case where all the type of density of boxes for each customer (see column Matheuristic of Table 1). In this paper, the proposed algorithm considers infeasible solutions, iteratively by worsening the transportation costs until reaching a feasible solution of minimum cost packing the demand of customers of the vehicles.

Table 1 Comparison with the best literature results

Instance	Nodes	Items	Vehicles	BKS	[15]		[17]		[18]		[24]		[19]		[29]		Mathuristic				
					Solution	Time	Solution	Time	Best	Average	Time	Best	Average	Time	Best	Average	Time	Solution	Time	Solution	Time
1	15	32	4	302.02	316.32	1800	321.47	13.2	304.13	305.35	12	302.02	302.02	72.3	303.21	98.85	300.69	492.9	296.03	107	
2	15	26	5	334.96	350.58	1800	334.96	11.5	334.96	334.96	0.6	334.96	334.96	0.9	334.96	4.55	359.78	941.1	292.17	7	
3	20	37	4	381.37	447.73	1800	430.95	540.6	399.68	409.79	121.8	392.63	401.44	182	398.05	93.86	374.81	228.7	347.69	154	
4	20	36	6	437.19	448.48	1800	458.04	323.5	440.68	440.68	5.4	437.19	437.54	16.1	440.68	46.75	430.88	492.9	360.91	8	
5	21	45	6	436.48	464.24	1800	465.79	99.6	450.93	453.19	30.9	443.61	451.03	183	452.56	63.98	495.32	495.3	495.32	3600	
6	21	40	6	498.16	504.46	1800	507.96	1212	498.32	501.47	18.4	498.16	498.38	23.6	498.56	196.9	845.51	3600	740.15	335	
7	22	46	6	767.46	831.66	1800	796.61	364.8	792.13	797.47	67.4	769.68	772.49	133	790.23	317.02	942.57	3600	752.37	496	
8	22	43	6	804.75	871.77	1800	880.93	230	820.67	820.67	78.6	810.89	821.35	139	820.67	98.9	840.44	3600	840.44	3600	
9	25	50	8	630.13	666.1	1800	642.22	982.2	635.5	635.5	16.3	630.13	645.81	24.3	635.5	353.07	639.25	3600	906.2	3600	
10	29	62	8	820.35	911.16	3600	884.74	1308	840.75	841.12	246.7	820.35	827.29	175	836.21	410.9	906.20	3600	840.44	3600	
11	29	58	8	772.85	819.36	3600	873.43	522.5	818.87	821.04	199.8	803.61	815.62	136	825.75	197.76	841.3	3600	641.3	3600	
12	30	63	9	614.59	651.58	3600	624.24	294.6	626.37	629.07	48.2	614.59	630.46	14	626.59	89.47	641.3	3600	641.3	3600	
13	32	61	8	2608.7	2928.34	3600	2799.74	2193	2739.8	2739.8	308.8	2646	2694.81	268	2739.8	319.78	2894.33	3600	2894.33	3600	
14	32	72	9	1368.4	1559.64	3600	1504.44	4581	1466.8	1472.26	642.8	1368.4	1413.59	312	1469.38	268.39	1673.19	3600	1673.19	3600	
15	32	68	9	1341.1	1452.34	3600	1415.42	2528	1367.6	1405.48	656.8	1341.1	1355.5	312	1369.69	356.55	1498.51	3600	1498.51	3600	
16	35	63	11	698.61	707.85	3600	698.61	4257	698.92	698.92	14.8	698.61	705.05	3.4	703.15	431.74	731.46	3600	731.46	3600	
17	40	79	14	866.4	920.87	3600	872.79	2096	868.59	870.33	14.9	866.4	917.96	2.5	872.05	374.84	881.53	3600	881.53	3600	
18	44	94	11	1207.7	1400.52	3600	1296.59	2275	1255.6	1261.07	2209.8	1207.7	1228.98	310	1250.86	325.74	1415.46	3600	1415.46	3600	
19	50	99	12	741.74	871.29	7200	818.68	2509	777.18	781.29	623.6	741.74	753.87	417	780.37	1374.8	843.4	3600	843.4	3600	
20	71	147	18	587.95	732.12	7200	641.57	1941	604.28	611.26	3901	587.95	596.42	427	605.59	1337	659.84	3600	659.84	3600	
21	75	155	17	1086.2	1275.2	7200	1159.72	2823	1110.1	1124.55	5180.6	1090.2	1107	443	1119.45	1247.9	1181.55	3600	1181.55	3600	
22	75	146	18	1147.8	1277.94	7200	1245.35	2686	1194.2	1197.43	2290.3	1147.8	1171.49	424	1167.28	1294.6	1242.84	3600	1242.84	3600	
23	75	150	17	1127.9	1258.16	7200	1231.92	4659	1158.5	1171.77	3727.6	1130.5	1135.46	426	1171.77	1105.7	1159.44	3600	1159.44	3600	
24	75	143	16	1114.1	1307.09	7200	1201.96	4854	1136.8	1148.7	1791.5	1116.1	1128.82	411	1136.27	2001.1	1222.98	3600	1222.98	3600	
25	100	193	22	1407.4	1570.72	7200	1457.46	5726	1429.6	1436.32	8817.1	1407.4	1428.8	453	1426.34	1458.8	1507.89	3600	1507.89	3600	
26	100	199	26	1585.5	1847.95	7200	1711.93	6283	1611.8	1616.99	6904.3	1600.4	1625.31	431	1585.46	3354.7	1832.86	3600	1832.86	3600	
27	100	198	23	1529.9	1747.52	7200	1646.44	9916	1540.7	1573.5	10484	1529.9	1550.85	435	1542.18	3140.2	1647.88	3600	1647.88	3600	
Average (Total Distance)					934.06	1042.26		997.18		966.67		953.79		960.10		941.59		1024.75		1001.08	
Relative GAP % [3L-VRP]					-1.68			2.76		6.01		7.44		6.73		8.83		-		-	
Relative GAP % [3L-CVRP]					-3.95			0.39		3.56		4.96		4.27		6.32		-		-	
Average Time [s]					4200		2416		1793		229		754		1101		2913		2577		
CPU					Pent. IV 3.0		Pent. IV 2.8		Pentium IV 3.2 GHz		Core2Duo E8500		Pentium IV 2.3		Pentium IV 2.67 GHz		i7-2600S		i7-2600S		17-2600S 2.86GHz
Single Thread Rating					653		635		699		1321		564		608		1797		1797		1797

6. Concluding remarks and future research

In this paper, a successful matheuristic algorithm has been proposed for solving the 3L-CVRP. The hybrid methodology decomposes the 3L-CVRP into two subproblems: the Capacitated Vehicle Routing Problem (CVRP) and the Three-dimensional Container Loading Problem (3D-SLOPP). The proposed approach combines a branch-and-cut algorithm for solving the vehicle routing problem (CVRP) with a GRASP approach in order to find a feasible solution for the 3D-SLOPP. The proposed algorithm has been compared with [15, 17-19, 24, 29] on the classical set of benchmark instances proposed for the 3L-CVRP. The results show the effectiveness of the proposed approach (some new best known solutions are found).

For future research, we will consider other mathematical formulations (i.e. three index mathematical formulation) that can be decomposed by exploiting the benefits of a branch-and-cut technique or a generation of columns. This consideration has advantages for solving the CVRP due to provides the control of the routes individually. In addition, it is possible to remove easily the capacity and subtour elimination constraints. However, it is necessary to make a careful treatment of this model because it implies an increase remarkable of the number of variables.

7. References

1. P. Augerat *et al.*, "Computational results with a branch and cut code for the capacitated vehicle routing problem", Université Joseph Fourier, Grenoble, France, Tech. Rep. 949-M, 1995.
2. P. Toth and D. Vigo, *The Vehicle Routing Problem*, Monographs on Discrete Mathematics and Applications, Philadelphia, USA, 2002.
3. B. Golden, S. Raghavan and E. Wasil, *The Vehicle Routing Problem: Latest Advances and New Challenges*. USA: Springer, 2008.
4. R. Baldacci, P. Toth and D. Vigo, "Exact algorithms for routing problems under vehicle capacity constraints", *Annals of Operations Research*, vol. 175, no. 1, pp. 213-245, 2010.
5. V. Schmid, K. Doerner and G. Laporte, "Rich routing problems arising in supply chain management", *European Journal of Operational Research*, vol. 224, no. 3, pp. 435-448, 2013.
6. A. Lodi, S. Martello and M. Monaci, "Two-dimensional packing problems: A survey", *European Journal of Operational Research*, vol. 141, no. 2, pp. 241-252, 2002.
7. V. Paschos, *Paradigms of Combinatorial Optimization: Problems and New Approaches*. John Wiley & Sons, 2013.
8. G. Wascher, H. Haubner and H. Schumann, "An improved typology of cutting and packing problems", *European Journal of Operational Research*, vol. 183, no. 3, pp. 1109-1130, 2007.
9. E. Coffman, J. Csirik, G. Galambos, S. Martello and D. Vigo, "Bin Packing Approximation Algorithms: Survey and Classification", in *Handbook of Combinatorial Optimization*, 2nd ed., P. Pardalos, D. Du and R. Graham (eds): New York, USA: Springer, 2013, pp. 455-531.
10. M. Alonso, R. Alvarez, J. Tamarit and F. Parreño, "A reactive GRASP algorithm for the container loading problem with load-bearing constraints", *European Journal of Industrial Engineering*, vol. 8, no. 5, pp. 669-694, 2014.
11. D. Álvarez, F. Parreño, R. Álvarez and F. Parreno, "A grasp algorithm for the container loading problem with multi-drop constraints", *Pesqui. Oper.*, vol. 35, no. 1, pp. 1-24, 2015.
12. L. Junqueira, J. Oliveira, M. Carravilla and R. Morabito, "An optimization model for the vehicle routing problem with practical three-dimensional loading constraints", *International Transactions in Operational Research*, vol. 20, no. 5, pp. 645-666, 2013.
13. A. Moura and J. Oliveira, "An integrated approach to vehicle routing and container loading problems", *OR Spectrum*, vol. 31, no. 4, pp. 775-800, 2009.
14. A. Moura, "A multi-objective genetic algorithm for the vehicle routing with time windows and loading problem", in *Intelligent Decision Support*, 1st ed., A. Bortfeldt, J. Homberger, H. Kopfer, G. Pankratz and R. Strangmeier (eds). Frankfurt, Germany: Springer, 2008, pp. 187-201.
15. M. Gendreau, M. Iori, G. Laporte and S. Martello, "A tabu search algorithm for a routing and container loading problem", *Transportation Science*, vol. 40, no. 3, pp. 342-350, 2006.
16. Y. Tao and F. Wang, "A new packing heuristic based algorithm for vehicle routing problem with three-dimensional loading constraints", in *Conference on Automation Science and Engineering (CASE)*, Toronto, Canada, 2010, pp. 972-977.
17. C. Tarantilis, E. Zachariadis, and C. Kiranoudis, "A hybrid metaheuristic algorithm for the integrated vehicle routing and three-dimensional container-loading problem", *IEEE Transactions on Intelligent Transportation Systems*, vol. 10, no. 2, pp. 255-271, 2009.
18. G. Fuellerer, K. Doerner, R. Hartl and M. Iori, "Metaheuristics for vehicle routing problems with three-dimensional loading constraints", *European Journal of Operational Research*, vol. 201, no. 3, pp. 751-759, 2010.
19. Q. Ruan, Z. Zhang, L. Miao and H. Shen, "A hybrid approach for the vehicle routing problem with three-dimensional loading constraints", *Computers & Operations Research*, vol. 40, no. 6, pp. 1579-1589, 2013.
20. L. Miao, Q. Ruan, K. Woghiren and Q. Ruo, "A hybrid genetic algorithm for the vehicle routing problem with three-dimensional loading constraints", *RAIRO - Operations Research*, vol. 46, no. 1, pp. 63-82, 2012.
21. W. Zhu, H. Qin, A. Lim and L. Wang, "A two-stage tabu search algorithm with enhanced packing heuristics for the 3l-cvrp and m3l-cvrp", *Computers & Operations Research*, vol. 39, no. 9, pp. 2178-2195, 2012.
22. G. Koloch and B. Kaminski, "Nested vs. joint optimization of vehicle routing problems with three-dimensional loading constraints", *Engineering Letters*, vol. 18, no. 2, pp. 193-198, 2010.
23. A. Bortfeldt and J. Homberger, "Packing first, routing second - a heuristic for the vehicle routing and loading

- problem", *Computers & Operations Research*, vol. 40, no. 3, pp. 873-885, 2013.
24. A. Bortfeldt, "A hybrid algorithm for the capacitated vehicle routing problem with three-dimensional loading constraints", *Computers & Operations Research*, vol. 39, no. 9, pp. 2248-2257, 2012.
 25. T. Feo and M. Resende, "A probabilistic heuristic for a computationally difficult set covering problem", *Operations Research Letters*, vol. 8, no. 2, pp. 67-71, 1989.
 26. M. Resende and C. Ribeiro, "Greedy Randomized Adaptive Search Procedures", in *Handbook of Metaheuristics*, 1st ed., F. Glover and G. Kochenberger (eds). Norwell, USA: Kluwer Academic Publishers, 2003, pp. 219-249.
 27. F. Parreño, R. Álvarez, J. Tamarit and J. Oliveira, "Neighborhood structures or the container loading problem: A VNS implementation", *Journal of Heuristics*, vol. 16, no. 1, pp. 1-22, 2010.
 28. R. Martí and J. Moreno, "Métodos multiarranque", *Inteligencia Artificial*, vol. 7, no. 19, pp. 49-60, 2003.
 29. Y. Tao and F. Wang, "An effective tabu search approach with improved loading algorithms for the 3l-cvrp," *Computers Operations Research*, vol. 55, pp. 127-140, 2015.